SAFEGUARD

An Assured Safety Net Technology for UAS

Evan T. Dill, Steven D. Young, and Kelly J. Hayhurst
Safety-Critical Avionics Systems Branch
NASA Langley Research Center
Hampton, VA, 23681

***Abstract*—As demands increase to use unmanned aircraft systems (UAS) for a broad spectrum of commercial applications, regulatory authorities are examining how to safely integrate them without loss of safety or major disruption to existing airspace operations. This work addresses the development of the Safeguard system as an assured safety net technology for UAS. The Safeguard system monitors and enforces conformance to a set of rules defined prior to flight (e.g., geospatial stay-out or stay-in regions, speed limits, altitude limits). Safeguard operates independently of the UAS autopilot and is strategically designed in a way that can be realized by a small set of verifiable functions to simplify compliance with regulatory standards for commercial aircraft. A framework is described that decouples the system from any other devices on the UAS as well as introduces complementary positioning source(s) for applications that require integrity and availability beyond what the Global Positioning System (GPS) can provide. Additionally, the high level logic embedded within the software is presented, as well as the steps being taken toward verification and validation (V&V) of proper functionality. Next, an initial prototype implementation of the described system is disclosed. Lastly, future work including development, testing, and system V&V is summarized.**

Keywords—assured containment; geo-fencing; Unmanned Aircraft System, formal methods; UAS Traffic Management (UTM)

I. INTRODUCTION

The recent advancement of UAS related technologies has caused a substantial increase in the desired use of such vehicles over the past few years. Entities in the public and private sectors around the world have begun to discover the potential benefits of UAS operations and are lobbying regulatory authorities to develop procedures to allow for more widespread operation. Until recently, commercial use of UAS in the United States (US) has been authorized primarily under Section 333 exemptions to the Federal Aviation Regulations (FARs) [1]. In June 2016, the Federal Aviation Administration (FAA) released Part 107 containing regulations to be included in the FARs for commercial use of small UAS [2]. The regulations under Part 107 cover UAS (a) weighing less than 55 lb, (b) operating in visual line-of-sight, and (c) operating under 400 ft above ground level or within 400 ft of a structure. UAS operating under Part 107 are prohibited from flying over any persons not directly participating in the operation.

While the steps taken under Part 107 enable a considerable number of UAS operations, additional regulations are needed for widespread use of larger, more capable UAS. Regulatory authorities around the world are working to broaden their regulatory frameworks for the rapidly growing and evolving UAS market. In doing so, authorities must strive to meet public demands for affordability without compromising their responsibilities to protect citizens, citizen's rights, and property; while also not crippling the UAS market through over-regulation.

Regulatory authorities in the commercial aviation sector have created a complex set of rules over many decades that provide high assurance with respect to safety while still satisfying the needs of multiple stakeholders. However, many of the methods that are used to comply with these rules (e.g., redundant systems, complex software and hardware certification processes, radar and human surveillance) would burden the UAS industry due to their high costs. Thus, many of the strategies used for achieving high levels of safety on commercial aircraft are not practical for the UAS market. Furthermore, while several research efforts have been undertaken to develop aviation-grade systems for UAS, little to no success has been achieved in terms of meeting reliability standards of conventionally piloted aircraft (CPA) [3]-[6]. This creates a precarious situation wherein an expedient strategy for integrating UAS into the National Airspace System (NAS) is desired, but the means to do so properly do not currently exist.

As an incremental step, the FAA has suggested "developing design standards tailored to a specific UAS application and proposed operating environment" [7]. This approach is consistent with the European Aviation Safety Agency's (EASA's) operations-centric framework for UAS operations [8]. Based on these regulatory approaches, Hayhurst et al. recently performed a case study to develop design standards for a 1000-lb unmanned rotorcraft with specific consideration for its operation in a rural agricultural environment [9]. At the heart of the aforementioned work, the authors establish two ideas that are fundamental to the research proposed in this paper.

Assertion #1: For a UAS, hull loss, in itself, is not a catastrophic failure condition with respect to safety.

For a CPA, the pilot, crew and passengers must be protected to avoid serious injury or loss of life. Consequently, hull loss is considered a catastrophic event for CPA. Furthermore, the hull of the aircraft must be preserved, as the vehicle itself is a valuable asset. In contrast, hull loss for UAS

does not inevitably endanger human life or property. The safety-related consequences of hull loss or loss of control for UAS largely depend on the environment in which the UAS operates. There are many possible UAS missions intended for sparsely populated or remote areas that pose little safety risk to people and property. Additionally, some unmanned vehicles are designed to be frangible or disposable, and many are relatively inexpensive. Therefore, prevention of hull loss is not always a substantial concern with respect to safety or economic loss. However, hull loss of a UAS is a safety concern with respect to collateral damage to other aircraft and to people and property on the ground. Under Assertion #1, the primary safety concern for UAS is not protecting the hull, but rather it is preventing catastrophic collateral damage.

Assertion #2: If UAS operations are kept within areas that minimize risk of collateral damage by minimizing exposure, safety standards and procedures can be simplified and minimized.

The Section 333 process and Part 107 are codified examples of this assertion. Both use extensive operational limitations to minimize risk of collateral damage, and consequently relieve approved applicants of requirements typical of airworthiness certification. This approach partitions the hazard space: one partition for hazards outside of the approved operational area, and the other for hazards within the operational area. This approach works well for missions where the operational area is confined to uninhabited, often low-altitude, environments. In such environments, hull loss is an economic concern, instead of a safety concern. Therein lies the opportunity to reduce the effort needed to establish airworthiness criteria for UAS built for such operations, as long as the method of containment to the operational area is highly reliable. In that case, many of the typical airworthiness requirements necessary to protect the physical vehicle are not necessary for the safety of the UAS. The focus of airworthiness standards in effect shifts from protection of the air vehicle to a focus on the system that ensures flights remain within the approved operational area (i.e., assured containment). This may allow the use of commercial-off-the-shelf (COTS) components for many UAS systems.

Geo-fencing is one method commonly deployed to contain UAS within a specific operational volume or stay-in region [10][11]. Most current geo-fencing techniques, such as those in commercially available autopilots, are effective in many scenarios, thus serving as a first line of defense against a breach of a stay-in region. However, neither commercial autopilots nor the geo-fences are typically developed in compliance with conventional certification standards for safety-critical systems. As a result, the reliability and dependability of these geo-fencing systems are unknown (at best) and inadequate (at worst). Most geo-fences reside on the same processor as the UAS's autopilot and share the same data sources. This lack of independence creates a single point of failure, whereby a hardware failure can cause a failure in both the autopilot and the geo-fencing function. Likewise, a software failure or data error in the geo-fence could lead to undesirable behavior by the autopilot. Stevens and Atkins provide a good discussion of how a geo-fencing system could be employed independent of the autopilot, and how different guidance modes can be defined to "replace the default guidance system" when a boundary violation is detected [12].

In addition to independence from the autopilot, the core functionality of a containment system depends on a reliable geo-referenced position estimate. Reliance on the Global Positioning System (GPS), or any combination of Global Navigation Satellite Systems (GNSS), can create another single point of failure. These space-based radio frequency (RF) systems share many common failure modes due to issues such as multipath, signal attenuation, and shadowing, and therefore, cannot be reliably depended on in a standalone manner.

To mitigate these issues in the context of our two safety assertions, the research presented here focuses on the development of a system, called Safeguard, that can provide a means of enforcing geospatial stay-in and stay-out regions while also achieving the reliability and dependability needed to satisfy conventional certification standards for CPA. The Safeguard system does not rely on other electronics on a UAS, is based on a minimal number of highly assured functions, and can be tailored to meet different assurance levels. From an integration standpoint, Safeguard can be easily ported to virtually any vehicle with almost no change to the UAS. Additionally, complementary positioning sources may be included as a means to mitigate hazards whose cause is sole reliance on GPS.

The following section lays out the Safeguard design framework, including its hardware mechanization, description of high level logic, and the inclusion of non-GNSS based positioning data. Additionally, the steps being taken toward meeting high assurance requirements are discussed. Subsequently, the current Safeguard system prototype and the results of initial flight tests are presented as a proof of concept. Finally, future work including plans for further development, testing, and system V&V are discussed.

II. Conceptual design and Operational concept

Safeguard is designed to monitor and predict non-conformance to a set of operating constraints. The current prototype device monitors and predicts non-conformance with geospatial stay-in and stay-out regions. Extensions of the prototype to include other mission-specific constraints such as maximum path deviation, speed limits, altitude limits is planned. The Safeguard device is isolated and independent of the unmanned aircraft's autopilot and operating system, and is easily implemented on most COTS UAS. The current design requires no inputs from any other onboard systems during flight and produces two outputs. The first output provides a warning to the autopilot of predicted violations to allow the autopilot an opportunity to change course. The second output terminates the flight if the vehicle does not respond adequately to the warning. The termination signal may be used in various ways (e.g., with different contingency maneuvers) depending on mission and safety requirements. Safeguard is agnostic to line-of-sight, does not require a command and control link, and can be configured without sole reliance on GPS. Lastly, the unit is designed to be small, lightweight, low-power, and

independently powered. The current target size, weight, and power (SWAP) is approximately 1”x2”x3”, 8 oz, and 300 mA (See Figure 1).

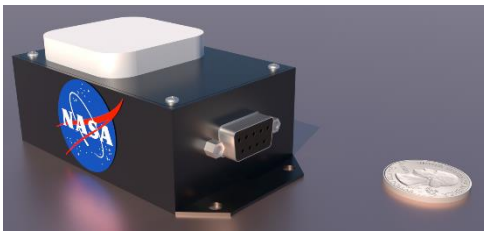

Figure 1. Safeguard Target Form Factor.

As will be discussed further in Section III, Safeguard has been designed from the outset to be a highly reliable system that can comply with aerospace standards for safety-critical systems. Compliance with system safety standards is a means within the aviation industry of assuring that a system will perform its intended function. For Safeguard, that means that the unmanned aircraft (UA) will always stay within its prescribed operational area (stay-in region) and out of specified no-fly zones. The system does not, however, provide any assurance that the UAS will perform its intended mission while operating within its geospatial constraints. Safeguard is strictly an independent safety system. Such a safety system may facilitate airworthiness certification of UAS with non-aviation-grade components that might pose challenges for conventional certification.

In Safeguard, preflight constraints and termination policies can be defined and implemented by the operator or, in the future, by a service provider, by an established database of geospatial constraints, or by some combination of those. In the current prototype, the UAS operator identifies and loads the Safeguard unit with the desired constraints (e.g., stay-in and stay-out boundaries), buffer criteria (e.g., warning times), and vehicle dynamics parameters prior to flight. A flight plan may also be loaded if Safeguard is tasked with monitoring flight plan excursions beyond some threshold. In the future, pre-flight information may come from a service provider, similar to the current digital notice to airmen (D-NOTAM) system that delivers no-fly zone information to manned aircraft and airline operations centers, or it may come from established databases of no-fly zones comparable to navigation databases used by CPA today.

The termination policy may also be an operator decision or an airspace management decision. In other words, once Safeguard detects that a violation is imminent (i.e., the autopilot has not responded to a warning), then the action to be taken can vary based on an operator-defined policy or a regulatory policy that governs the airspace. Regardless, these policies must be established prior to flight so that appropriate standards can be applied. Safeguard is not designed to make decisions during flight as to best course of action in a given context. However, a complementary research effort is addressing this aspect [13].

III. Methodology and Framework for Detecting Boundary Violations

To enforce conformance to geospatial constraints, the Safeguard system establishes three boundaries for each stay-in and stay-out region: a hard boundary, a soft boundary, and a warning boundary, as shown in Figure 2 for simple square areas. The hard boundary, shown in red, is a user-defined polygon representing a geospatial region that should never be breached. Safeguard allows polygons to be virtually any shape or size, as long as the polygon is closed. The associated soft and warning boundaries follow the shape of the hard boundary with buffers based on the vehicle dynamics. The points that define the hard boundary are loaded prior to flight and do not change throughout a mission. It is considered a system failure if Safeguard does not prevent the UAS from violating any hard boundary.

The soft boundary, indicated in yellow in Figure 2, specifies when a flight termination maneuver is necessary to ensure that the hard boundary is not violated. If the UA crosses the soft boundary, it is assumed that a loss of control or unrecoverable fly away has occurred; that is, the means to safely control a UA have been compromised in some manner. Therefore, flight termination is the only reliable method for preventing excursions into no-fly zones.

To prevent any breaches of the hard boundaries, the Safeguard system computes the maximum distance that a UA could travel once flight termination is initiated. This distance is denoted as ε. Establishing the soft boundary at a distance ε from all points along the hard boundary ensures that a UA maintains a minimum safe separation distance (MSSD) from the hard boundary such that flight termination would prevent violations of no-fly zones. As the flight trajectory resulting from a flight termination signal is situational, the shape and size of soft boundaries dynamically change during flight. Changes are based on the current state of the vehicle, a set of parameters classifying the vehicles aerodynamics, as well as wind and weather conditions to some extent. Rudimentary methods for calculating the MSSD between a UAS and all hard boundaries are discussed in more detail later in this section and research toward more accurate methods for computing ε is discussed in Section VI.

The final boundary, shown in green in Figure 2, is the warning boundary. The warning boundary defines the points when a notice will be issued to systems on a UAS (e.g., the autopilot) that a vehicle’s current state is approaching a soft boundary. While no direct action is commanded based on this warning, it allows the UAS to attempt to perform a contingency maneuver to avoid flight termination. The warning boundary dynamically changes as a function of ε multiplied by a tunable scale factor ρ ($\rho>1$). The scalar ρ is a tunable parameter to provide operators flexibility with respect to their desired proximity to the soft boundary where the termination signal will be generated. The operator can establish and load an appropriate scale factor such that Safeguard issues warnings at a distance of $\rho\varepsilon$ from all points along the hard boundary.

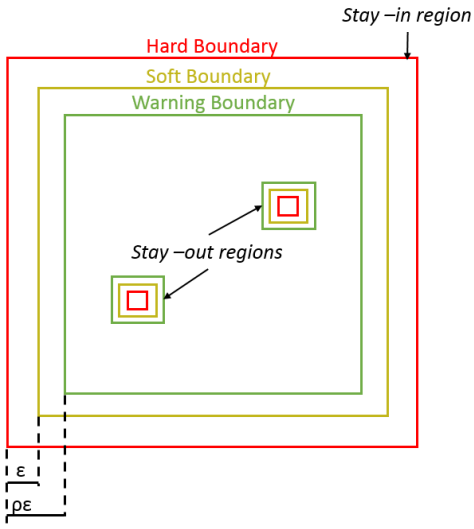

Figure 2. Conceptual Boundaries for Geospatial Constraint Conformance.

To create a system capable of reliably enforcing compliance with the aforementioned boundaries while simultaneously meeting other previously mentioned requirements, the system architecture shown in Figure 3 was derived. The hardware components consist of (a) a processor; (b) a GPS/Inertial Navigation System (INS)/altimeter unit; (c) an optional alternative non-GPS positioning, navigation, and timekeeping (PNT) system; (d) an input port; and (e) three output ports. Prior to flight, boundary points designating stay-in and stay-out regions, vehicle dynamics parameters, and an optional flight plan are loaded into Safeguard. This information can be manually entered by operators or it can be obtained through a service provider such as the UAS traffic management (UTM) system [14], which is currently under development. At start-up, the termination and warning lines default to a "violation" state and are only set to a "compliant" state when the system receives sufficient proof that the vehicle's state is compliant with all conformance criteria. This prevents the UAS from taking off until the Safeguard system has been properly initialized and deems the vehicle's current state to be safe for operation with respect to the conformance criteria.

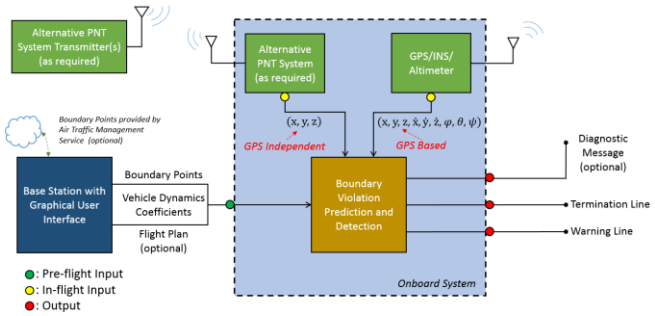

Figure 3. System Architecture *(Figure will be expanded across both columns in final format.)*

Prior to flight, multiple inputs are required to ensure intended functionality of the Safeguard system. Those inputs include hard boundary points to indicate all no-fly zones and vehicle dynamics coefficients to represent the aerodynamics model of the UAS. If a flight plan is loaded, Safeguard can check for violations of the intended path with any geospatial constraints, or it can monitor for excessive flight plan deviations if desired. Due to the functional importance of valid and correct geo-referenced hard boundary points, all such data are captured, processed, and transferred in accordance with appropriate Data Processing Assurance Levels (DPAL) as defined in [15] for similar types of data used on commercial aircraft (e.g., for navigation data). These points can either be obtained from an authorized service provider, or through manual input of surveyed locations captured by the operator or from qualified professionals.

Once properly initialized, the Safeguard system begins monitoring to detect any breaches of the defined boundaries. With each sample of the PNT data, the warning and terminate lines are set to either "compliant" or "violation", accordingly. To properly interpret the meaning of these two signals, these lines usually are connected to two separate elements of the UAS. The warning line is typically connected to a system that has control authority of the aircraft, such as the autopilot, to allow that system to initiate a contingency maneuver to avoid flight termination. Because off-the-shelf UAS systems may fail, Safeguard's ability to command flight termination is independent of other systems on-board the UAS. The terminate line is connected to a separate termination mechanism that complies with appropriate airworthiness standards. Finally, a message containing diagnostic information is output via a serial connection from the system. This message is not necessary for proper operation of the Safeguard system, but it can be used by the operator to monitor the state of the aircraft and the state of the Safeguard system for situational awareness. All outputs coming from the Safeguard system are optically isolated to inhibit any signals from entering the system during operations.

A. Calculation of a Minimum Safe Separation Distance

To assure containment based on the boundary definitions shown in Figure 2, it is necessary to calculate a MSSD between the hard boundary and the soft boundary. In the current prototype, the MSSD is calculated and updated during flight, based on the vehicle's state and a small set of parameters to describe the vehicle's aerodynamic characteristics. Currently, the vehicle's state is defined as its current geo-referenced position, $\mathbf{x} = [x, y, z]$, velocity, $\mathbf{v} = [v_x, v_y, v_z]$, and attitude, $\mathbf{\Psi} = [\varphi, \theta, \psi]$, and a set of vehicle dynamics parameters: mass, spatial area of the wings, nominal lift to drag ratio, degraded lift to drag ratio (after termination is initiated), lift coefficient, and drag coefficient. In future iterations of the Safeguard system, metrics to describe wind conditions will be included.

Results from simulations of the MSSD calculation are shown below for a ~10 lb multi-rotor in Figure 4 and a small fixed wing UAS in Figure 5. These results represent the possible flight trajectories of the two UAS after a termination is initiated. As a multi-rotor can change directions quickly, its possible trajectories are all nearly equidistant from the UAS's

current location. The possible trajectories of a fixed wing aircraft form a fan pattern symmetric with respect to the aircraft's velocity vector. In scenarios where control authority of the aircraft is lost, the trajectories shown below represent the typical behavior of their respective platforms.

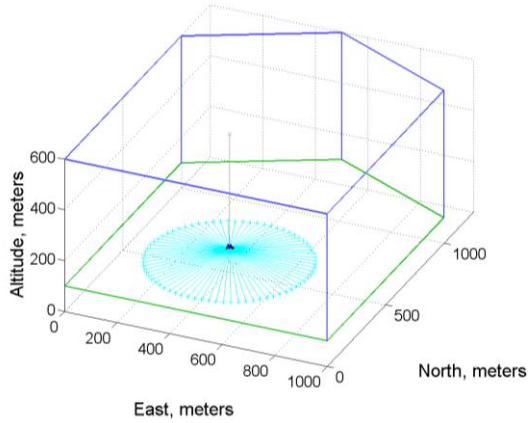

Figure 4. Modeled Trajectory of a Multi-Rotor UAS after Flight Termination When Hovering at 400 meters.

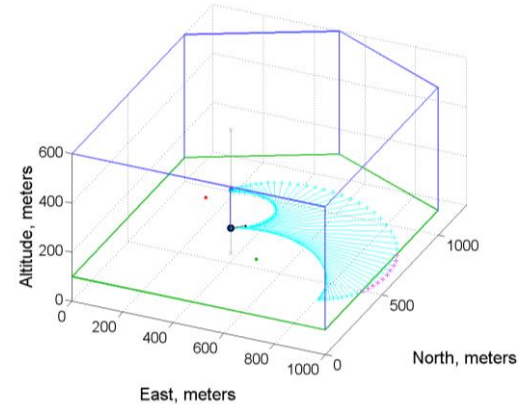

Figure 5. Modeled Trajectory of a Fixed Wing UAS after Flight Termination When Flying at 400 meters and 80 knots in a Northeast Direction.

IV. Assurance Approach

One of the unique and essential attributes of the Safeguard concept is that it is designed to meet stringent standards such as those established for safety-critical systems on commercial aircraft and/or spacecraft. The intent is to provide a certifiable system consistent with Proposal 6 in EASA's UAS framework "to prevent unintended flight outside safe areas and to increase compliance to applicable regulation" [8]. The following are the four main elements of the Safeguard system being addressed in the research, development, and testing of prototypes:

1. Positioning system
2. Boundary Database (e.g., hard boundaries)
3. Boundary Monitoring and Violation Detection Software
4. Termination mechanism

Our approach to assurance for each of the four system elements is briefly described below. An initial set of hazards and considerations for mitigation are provided in the Appendix.

A. Positioning System Performance

For the Safeguard system to reliably perform its monitoring function, accurate and timely position estimates are critical. For CPA, positioning data is available from a myriad of independent systems such as GPS, very high frequency omnidirectional range (VOR) stations, distance measuring equipment (DME), tactical air navigation (TACAN) stations, and high quality inertial navigation systems (INSs). Additionally, for CPA, redundancy is employed to mitigate any potential failures and ensure continuous operations. Unfortunately, UAS typically operate at low altitudes making ground-based radio frequency systems like VOR, DME and TACAN unobservable due to line of sight issues. Moreover, most UAS operate with sensors that are lightweight and relatively low cost. This typically results in a lower performing positioning system consisting of a GPS receiver and a lower-grade inertial measurement unit (IMU). For Safeguard to be an effective monitor, the performance of its positioning system must be better than the performance of the system embedded within the UAS.

For many UAS operational scenarios, GPS can provide adequate accuracy, availability, and continuity; particularly if applying well-established techniques for improving its performance such as receiver autonomous integrity monitoring (RAIM), the Wide Area Augmentation System (WAAS), or a satellite-based augmentation system (SBAS). However, for some safety-critical applications, GPS is not a viable stand-alone option. For example, when operating in environments such as urban canyons or around dense foliage, GPS signals may be unattainable or significantly deteriorated due to shadowing, signal attenuation, or multipath issues. Furthermore, GPS is inherently susceptible to intentional or unintentional jamming as well as sophisticated attacks such as spoofing. One common method to augment GPS in challenging environments is integration with an IMU [18]. This integration has been shown to increase positioning accuracy and availability as well as allow for operations to continue during short GPS outages. However, a GPS/IMU integration is still reliant on GPS signals for long term stability, and therefore, may not be sufficient for some missions. Due to these issues, a secondary independent positioning system can become necessary to augment GPS.

There are many active research and development efforts attempting to find a solution to this problem. These efforts include a wide array of ideas ranging from prominent techniques such as vision and laser-based navigation [19], to less pervasive ideas such as navigating off of magnetic fields [20] or using signals of opportunity [21]. Unfortunately, none of these concepts have as yet demonstrated the reliability and dependability needed for a safety-critical system such as Safeguard. Based on the amount of research being dedicated

to this problem and the importance of finding a solution, it is assumed that eventually there will be an alternative ubiquitous positioning system capable of serving as a viable backup to GPS. As such, we designed our architecture to be easily adaptable to new positioning systems in the future.

While an independent global positioning solution is not currently available, multiple local positioning systems (LPS) have demonstrated the ability to serve as backups for GPS within a localized region. To develop and test Safeguard's functionality without requiring sole reliance on GPS, a Locata® LPS [22] was chosen as the alternative PNT (APNT) system. The Locata® system uses a network of ground-based transceivers to provide geo-referenced positioning to mobile receivers within range of the radio frequency signals.

In summary, the Safeguard positioning system must perform well with respect to accuracy, integrity, availability, and continuity of service. How well will depend on the mission to be flown and the level of risk deemed acceptable with respect to violating prescribed constraints (e.g., hard boundaries). Given these requirements, the Safeguard positioning system architecture can be tailored to meet them, applying many of the techniques established for CPA.

B. Boundary Database Integrity

The second essential input to the detection algorithm is the set of constraints that are specified pre-flight and loaded onto the Safeguard unit. Of these, the most complex and vulnerable to errors are the polygons that represent the hard boundaries (i.e., no-fly zones). Fortunately, there are several established industry standards for assuring the content and quality of these types of data. These standards were established for commercial transport aircraft that utilize similar geospatial data for navigation and situation awareness systems, where probability of failure must be very low. For the Safeguard databases, we leverage and apply relevant aspects from four of these standards [15][23][24][25]. A fifth is also relevant and used as guidance [26]. A sixth is currently being revised and may also include relevant material [27]. Only the first four will be summarized here with respect to their role in Safeguard database assurance.

Standards for geospatial databases that contain data representing airport features, terrain, and/or obstacles are defined in [23] and [24]. Database elements are defined using points to represent the vertices of closed polygons. The standards specify general requirements (e.g., the spatial and temporal reference system) and specific rules for how to capture polygons, what types of polygons to capture, how and how often to update these data, and how good the data must be (i.e., data quality requirements). The feature types that are relevant to Safeguard are vertical objects, aerodrome structures, and terrain features. Each may be used as a basis for defining a hard boundary for a stay-in or stay-out region in Safeguard.

Standards for the exchange of geospatial databases are defined in [25]. Because data may come from multiple sources and may be used by multiple applications, a standard data model must be used to assure consistent interpretation. Safeguard will apply this model when data representing hard boundaries must be combined from multiple sources to create a complete set, or when interpreting data provided by others in accordance with the standard.

Standards for processing databases that are to be used onboard aircraft are defined in [15]. Any data to be acquired, processed, and loaded onto an aircraft system should comply with this standard, as well as guidance provided in [26]. The primary intents are to assure that (a) the data provided meets all of the requirements for its intended use, and (b) data has not been altered or corrupted since origination. Seven quality characteristics are established in [15] wherein evidence must be provided to support the claims of the designer with respect to meeting the system's data quality requirements. These are:

1. Accuracy – The degree of conformance between the estimated or measured value and its true value
2. Resolution – The number of units or digits to which a measured or calculated value is expressed and used
3. Assurance Level – The degree of confidence that a data element is not corrupted while stored or in transmission
4. Traceability – The degree that a system or a data product can provide a record of the changes made to that product and thereby enable an audit trail to be followed from the end-user to the data originator
5. Timeliness – The degree of confidence that the data is applicable to the period of its intended use
6. Completeness – The degree of confidence that all of the data needed to support the intended use is provided
7. Format – The structure of data elements, records and files arranged to meet standards, specifications or data quality requirements

For Safeguard, the requirements for six of these are given in [15][23][24][25] and are assumed to be sufficient for most missions. Characteristic #3 is referred to as the Data Processing Assurance Level (DPAL) and, per the standard, may be one of three levels (1, 2, or 3); with "1" being the highest degree of confidence. Typically, the DPAL will correspond to the Design Assurance Level (DAL) associated with the software that uses the database [16]. For example, a DPAL of "1" corresponds to a DAL of "A" and "B" (that is, software whose anomalous behavior could contribute to a catastrophic or hazardous failure condition).

As with positioning system performance, it is expected that the DPAL requirement for pre-loaded data in Safeguard will vary across missions and operating environments based on the level of risk deemed acceptable with respect to violating prescribed constraints (e.g., hard boundaries). For research and development purposes, we assume the most stringent will be required (DPAL 1). The method to achieve DPAL 1 will depend on whether the data originates locally via a process managed and performed by the operator, or the data is provided as a service from a certified source. Details on both of these methods will be published separately.

C. *Verification of Boundary Detection Software*

To support the creation of highly-assured algorithms, the boundary detection and violation recognition functions embedded within the Safeguard code have been developed and verified using formal methods [28]. Work is also in progress to develop the software in compliance with NASA's software safety requirements. In addition, Monte Carlo evaluations were conducted in Matlab to provide evidence of proper functionality of the boundary detection and violation algorithms. During these experiments, results were compiled and evaluated for each function used for boundary detection and violation based on a large set of inputs that were randomly generated across the spectrum of possibilities. An example of the results produced through Monte Carlo evaluations are shown below in Figure 6. In this example, functions designed to determine if a given point is inside or outside arbitrarily shaped polygons were evaluated. One stay-in region as well as three stay-out regions were created and 1,000,000 random points were assessed for compliance. Points deemed to be in compliance with all designated regions are shown in green, while points that are in violation of the regions are shown in red. Through evaluation of the results, it was observed that all of the generated points were properly identified as either in violation or compliant. Similar analyses were conducted for each formally verified function.

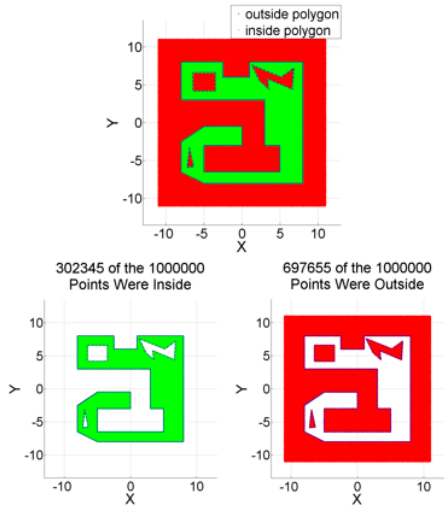

Figure 6. Monte Carlo Evaluation of Boundary Violation Function.

D. *Testing the Termination Mechanism*

Unlike other components of the Safeguard system, the mechanism to initiate a flight termination must be somewhat vehicle specific. The main drivers behind this are (1) the termination policy established by the operator or regulator; (2) the aerodynamic differences between rotorcraft and fixed wing aircraft; and (3) the propulsion differences between battery and gas-powered vehicles. Without the use of incendiaries, which is not considered a safe option in many cases, there is no termination system that is completely vehicle agnostic. Therefore, the Safeguard system has been designed to function with a few different termination strategies.

For testing to date, two termination policies have been used. Flight termination is achieved by discontinuing power to the motors, or for certain aircraft, forcing control surfaces into specific positions. To halt power to the motors of an electric vehicle, simply severing one of the battery cables will nearly instantaneously yield the desired results. This can be accomplished through the use of COTS electrically triggered cable cutters such as those designed to remotely diffuse explosives [29]. For a gas-powered vehicle, power can be denied to the motors by choking the fuel line and stopping the injection of gas into the motor.

In certain cases, such as a sailplane or flying wing, simply discontinuing power to the motors will not adequately terminate flight as the vehicle can glide for too great a distance. In these situations, it is desirable to force certain control surfaces into specific positions to expedite the termination process. For example, many fixed wing aircraft can be terminated quickly by commanding both ailerons into opposing maximum positions and the rudder to a maximum state. The inclusion of control surfaces in the termination process is typically only needed for fixed wing vehicles, but could be employed on multi-rotors if needed. The challenge, if control surfaces need to be included in the termination process, is to provide a function to actuate them independently of other UAS systems, including the autopilot and any primary wires, cables, and power sources.

V. Proof of Concept

To test the validity of the concepts underlying Safeguard, an initial prototype system was created. This prototype, shown integrated onto a multi-rotor in Figure 7, was designed with COTS hardware to rapidly evaluate the effectiveness of the Safeguard algorithms and methodology. The warning line was connected to the vehicle's autopilot, the terminate line was connected to a system which simply disconnected power to the UAS' motors, and the diagnostic message was wirelessly transmitted to a laptop base station.

Figure 7. Safeguard Prototype on a Multi-rotor UAS (2015).

Using this prototype Safeguard system, multiple test flights were conducted. To show that the Safeguard system is vehicle and autopilot agnostic, tests were completed on various rotorcraft platforms, each operating with dissimilar autopilots. Experiments were designed and conducted that resulted in vehicle contingency maneuvers as well as flight terminations.

These successful test flights demonstrated the effectiveness of the Safeguard system for assured geospatial containment within designated boundaries. Portions of the test flights can be viewed online [30].

A second prototype is currently being tested (Figure 8). The five primary objectives of this test phase are to: (1) demonstrate performance across additional vehicle types, including small fixed-wing UAS; (2) demonstrate functionality during periods of degraded GPS performance, including loss of GPS; (3) evaluate alternate termination strategies; (4) demonstrate integration with UTM services; and (5) test across two mission types. The first mission employs Safeguard as a monitor for test range excursions when a vehicle is operating beyond visual line-of-sight (i.e., serves as a virtual Range Safety Officer). The second mission employs Safeguard to monitor for safe stand-off distance while a UAS is inspecting power system transmission lines and associated structures. If the UAS gets too close to the transmission line or structures, Safeguard signals the UAS to retreat. This work is currently in preparation for publication.

Figure 8. Current Safeguard Prototype (2016).

VI. Future Work

To develop Safeguard to the desired DAL, much work still remains to be done. This work includes hardware development, improving the computational robustness of the MSSD estimation (i.e., the soft boundary), as well as system-level V&V. While a vast majority of the software currently implemented on Safeguard has been rigorously developed and scrutinized, several hardware components are COTS products with no associated reliability estimates. Moreover, little effort has been spent to minimize the size, weight, power and cost (SWAP-C) of the hardware. However, based on an analysis of the computing and sensor requirements for Safeguard, it is believed that a ruggedized version could be produced to the form factor shown in Figure 1. Reducing the SWAP-C and converting to ruggedized hardware will be necessary steps in the development of Safeguard. Steps are underway to achieve this via partnering discussions with manufacturers of similar devices, as well as using NASA in-house expertise.

To improve the MSSD estimation, work is underway to better characterize vehicle dynamics. We believe that loading a small set of vehicle dynamics parameters may be sufficient to characterize the vehicle for our purposes, rather than loading a high-fidelity aerodynamic model. While the current set of vehicle dynamics have been shown to adequately describe multiple UAS, additional testing is needed for a broader range of aircraft. Data captured during flight termination events may also help refine the MSSD estimation. As it is not practical to terminate a multitude of vehicles, a newly developed UAS simulation capability at NASA Langley is being leveraged. This simulation enables testing and refinement of different computational methods for determining the MSSD in an accelerated and benign fashion. The UAS simulator will allow for accelerated Monte Carlo evaluations across a span of input uncertainties and other failure modes without having to physically crash an abundance of UAS.

The other uncertainty that can affect the MSSD estimate is wind. One approach being explored is overbounding wind speeds and directions. Flights may also be restricted from takeoff if winds exceed a certain threshold.

Conducting test flights on various UAS in various environmental conditions (e.g. in wind conditions and in rugged terrain with GPS dropouts) will be a necessary part of V&V of the Safeguard device. Flight test data will be used to verify that the system works as designed, and will also serve to validate data used in the Monte Carlo simulations. Flight test data in conjunction with the data gathered during the development in compliance with NASA standards will be used to support a safety case for using Safeguard.

Finally, over the next 2-3 years, phased integration testing is planned with other complimentary systems including ICAROUS [13] and UTM [14] that are being developed concurrently. In a similar fashion to Safeguard, ICAROUS is being designed to monitor conformance to a set of rules in addition to performing collision avoidance with other cooperative (known) aircraft. These two systems complement each other as ICAROUS has the decision and control authority that Safeguard lacks, while Safeguard can provide the independent highly assured monitoring function that ICAROUS does not possess. The other system that Safeguard will be integrated with, UTM, is a service-oriented ground-based architecture that will provide for air traffic management of low altitude UAS operations, much like the current ATM system provides for safe CPA operations. As described previously, Safeguard can be configured to receive operational constraints (e.g., no-fly zones) from a UTM-like service. In addition, Safeguard could provide the "heartbeat" signal required by UTM for UAS operating in its managed airspace.

References

[1] Federal Aviation Administration, "Petitioning for exemption under Section 333," [Online] http://www.faa.gov/uas/legislative_programs/section_333/how_to_file_a_petition/, 2014.

[2] Federal Aviation Administration, (2013 July 26), "One giant leap for unmanned-kind," [Online] http://www.faa.gov/news/updates/?newsId=73118&omniRss=news_updatesAoc&cid=101_N_U, 2013.

[3] Clough, B., "Unmanned aerial vehicles: autonomous control challenges, a researcher's perspective," Journal of Aerospace Computing, Information, and Communication," vol. 2, pp. 327-347, 2005.

[4] United States Government Accountability Office, "Unmaned aircraft systems, federal actions needed to ensure safety and expand their potential uses within the National Airspace System," GAO-08-511, 2008.

[5] Williams, K., "A summary of unmmaned aircraft accident/incident data: human factors implications," DOT-FAA-AN-04-24, 2004.

[6] Gertler, J., “U.S. unmanned aerial systems, congressional research service report for Congress,” R42136, 2012.

[7] Federal Aviation Administration, “Integration of civil Unmanned Aircraft Systems (UAS) in the National Airspace System (NAS) roadmap,” US Department of Transportation, First edition, 2013.

[8] European Aviation Safety Agency, “Introduction of a regulatory framework for the operation of drones”, Advance Notice of Proposed Amendment 2015-10, July 31, 2015.

[9] Hayhurst, K., Maddalon, J., Neogi, N., Verstynen, H., “A Case Study for Assured Containment,” International Conference on Unmanned Aircraft Systeems”, 2015.

[10] Ardupilot, “Simple geofence,” [Online] http://copter.ardupilot.com/wiki/ac2_simple_geofence/.

[11] Atkins, E., “Autonomy as an enabler of economically-viable, beyond-line-of-sight, low-altitude UAS application with acceptable risk,” AUVSI unmanned Systems, 2014.

[12] Stevens, M. and Atkins E., “Multi-Mode Guidance for an Independent Miltocopter Geofecing System,” 16= AIAA Aviation Technology, Integration, and Operations Conference, June 2016.

[13] Consiglio, M., Munoz, C., “ICAROUS: Integrated Configurable Algorithms for Reliable Operations of Unmanned Systems”, IEEE Digital Avionics Systems Conference, 2016.

[14] Kopardekar, P., “Unmanned Aerial System (UAS) Traffic Management (UTM): Enabling Low Altitude Airspace and UAS Operations,” NASA Ames Technical Memorandum, 2014.

[15] "Standards for Processing Aeronautical Data," RTCA Document DO-200B, RTCA, June 2015.

[16] “Software Considerations in Airborne Systems and Equipment Certification,” RTCA Document DO-178C, RTCA, December 2011.

[17] “Design Assurance Guidance for Airborne Electronic Hardware,” RTCA Document DO-254, RTCA, April, 2000.

[18] Farrell, J., “GNSS Aided Navigation and Tracking,” American Literary Press, 2007.

[19] Dill, E., “GPS/Optical/Inertial Integration for 3D Navigation and Mapping Using Multi-copter Platforms,” Ph.D. Dissertation, Ohio University, 2014.

[20] Storms, W., Shockley, J., Raquet, J., “Magnetic Field Navigation in an Indoor Environment,” Ubiquitous Positiong Indoor Navigation and Location Based Services, 2010.

[21] McEllroy, J., Raquet, J., Temple, M., “Opportunistic Navigation: Finding Your Way with AM Signals of Opportunity”, GPS World, 2007.

[22] Rizos, C., “Locata: A Positinging System for Indoor and Outdoor Applications Where GNSS does not Work,” Proceedings of the 18th Association of Public Authority Surveyors Conference, 2013.

[23] "User Requirements for Aerodrome Mapping Information," RTCA Document DO-272D, RTCA, November 2015.

[24] "User Requirements for Terrain and Obstacle Data," RTCA Document DO-276C, RTCA, November 2015.

[25] "Interchange Standards for Terrain, Obstacle and Aerodrome Mapping Data," RTCA Document DO-291C, RTCA, November 2015.

[26] FAA Advisory Circular, “Acceptance of Aeronautical Data Processes and Associated Databases,” AC-20-1538, April 2016.

[27] “Standards for Aeronautical Information,” RTCA Document DO-201A, RTCA, April 2000.

[28] Monin, J., “Understanding Formal Methods,” Springer-Verlag, London, 2003.

[29] [Online] http://www.nexus-defence.com/shop/k-cutter-electronic-wire-cutter/

[30] [Online] https://www.youtube.com/watch?v=ljHfuC-GiEs.

[31] “Software Assurance Standard,” NASA Technical Standard NASA-STD-8739.8, NASA, 2004.

VII. Appendix: Preliminary List of Safeguard Hazards and Mitigation Considerations

No.	Hazards	Mitigation Considerations
1	Degraded, or loss of, positioning system data (e.g., loss of GPS, inaccurate estimates)	Employ RAIM, WAAS, SBAS, and/or GPS-independent APNT system as necessary to meet mission-specific requirements for accuracy, integrity, availability, and continuity of service; Leverage established/proven integration techniques (e.g. for GPS/INS/APNT) Use proven and flight-qualified connectors and cabling
2	Invalid constraint data (e.g., incorrect or missing survey data for hard boundaries)	Comply with industry standards for data capture, maintenance, processing and quality assurance [15][23][24] [25] as necessary to meet mission-specific requirements; Leverage existing authorized processes and products as able
3	Invalid or insufficient setting of warning and soft boundaries (e.g., incorrect assumptions regarding ballistic flight trajectory after killing power)	Buffers may not allow enough time for autopilot response or for flight termination to stay within boundary. Conservative setting of buffers based on Monte Carlo simulations and/or testing to limit the possibility of any collateral damage (e.g., parts or debris leaving the containment region). Simulations should evaluate across anticipated wind, sensor, and aerodynamic uncertainties; Constrain operations to an upper wind limit if necessary. Comply with standards for software assurance [16][31]; Use proven and flight-qualified operating system, memory, and central processing unit hardware
4	Failure to predict or inaccurate prediction of a boundary violation (e.g., missed detection or false alarm)	Apply formal methods to verify approach for determining stay-in/stay-out status. Comply with standards for software assurance [16][31]; Set required assurance level or software classification in accordance with acceptable missed detection rate Use proven and flight-qualified operating system, memory, and central processing unit hardware
5	Failure to command or inappropriate command for contingency action for warning violation (e.g., signal to the autopilot for contingency response)	Comply with standards for software and hardware assurance [16][17]. Use proven and flight-qualified operating system, memory, and central processing unit hardware Use proven and flight-qualified connectors and cabling
6	Failure to command or inappropriate command for flight termination	Termination mechanism shall comply with standards for software and hardware assurance [16][17]. Use proven and flight-qualified operating system, memory, and central processing unit hardware Use proven and flight-qualified connectors and cabling
7	Failure of or insufficient power (e.g., failure of the battery or any power source supporting the Safeguard unit or termination mechanism)	Use proven and flight-qualified battery with at least 3x margin with respect to expected flight time
8	Failure or degradation of the diagnostic link to the ground-based operator (e.g., for cases where Safeguard is to warn a remote pilot of a buffer violation)	Confirm that autopilot-based contingency maneuvers can be a fail-safe default when a warning is issued but communication link has failed